

Exhibit 10 to Complaint
Intellectual Ventures I LLC and Intellectual Ventures II LLC

Example Chase Count III Systems and Services
U.S. Patent No. 7,280,998 (“the ’998 Patent”)

The Accused Systems and Services include, without limitation, Chase’s systems that utilize Trino and/or Presto (“Trino/Presto”); all past, current and future systems and services that operate in the same or substantially similar manner as the specifically identified systems and services; and all past, current and future Chase’s systems and services that have the same or substantially similar features as the specifically identified systems and services (“Example Chase Count III Systems and Services”).

U.S. Patent No. 7,280,998	
Example Claim 1	Example Chase Count III Systems and Services
1. An arrangement for providing information to a plurality of different client views within an enterprise without creating an actual data warehouse, the arrangement comprising	<p>Upon information and belief, Chase's systems include "an arrangement for providing information to a plurality of different client views within an enterprise without creating an actual data warehouse, the arrangement comprising." Chase uses Trino/Presto for producing data warehousing analytics including producing reports. Trino/Presto perform data analytics without the data warehouse.</p> <p>What Trino is</p> <p><u>Trino is a tool designed to efficiently query vast amounts of data using distributed queries. If you work with terabytes or petabytes of data, you are likely using tools that interact with Hadoop and HDFS. Trino was designed as an alternative to tools that query HDFS using pipelines of MapReduce jobs, such as Hive or Pig, but Trino is not limited to accessing HDFS. Trino can be and has been extended to operate over different kinds of data sources, including traditional relational databases and other data sources such as Cassandra.</u></p> <p><u>Trino was designed to handle data warehousing and analytics: data analysis, aggregating large amounts of data and producing reports. These workloads are often classified as Online Analytical Processing (OLAP).</u></p> <p>See https://trino.io/docs/current/overview/use-cases.html (last accessed on November 10, 2023).</p> <p><u>Trino allows users to perform data warehouse analytics without the data warehouse, enabling users to query data anywhere it lives. In practice, an analyst doesn't need to load the data, transform the data, or do preparation of any kind. With this, an analyst can then access data anywhere, using regular SQL queries, without having to worry about the underlying infrastructure that makes it all work. This is an alternative to the traditional approach of collecting and consolidating data in a centralized data warehouse.</u></p> <p>See https://www.work-bench.com/post/trino-a-distributed-query-engine (last accessed on November 10, 2023).</p>

What Presto Is

Presto is a tool designed to efficiently query vast amounts of data using distributed queries. If you work with terabytes or petabytes of data, you are likely using tools that interact with Hadoop and HDFS. Presto was designed as an alternative to tools that query HDFS using pipelines of MapReduce jobs such as Hive or Pig, but Presto is not limited to accessing HDFS. Presto can be and has been extended to operate over different kinds of data sources including traditional relational databases and other data sources such as Cassandra.

Presto was designed to handle data warehousing and analytics: data analysis, aggregating large amounts of data and producing reports. These workloads are often classified as Online Analytical Processing (OLAP).

See <http://prestodb.io/docs/0.283/overview/use-cases.html> (last accessed on November 10, 2023).

Presto's power and value proposition is defined by the fact that it can query data wherever it is stored, without the need to move the data into a separate, structured system like a relational database or data warehouse. Finally, query execution runs in parallel over a scalable "pure memory-based architecture," returning results in seconds irrespective of the size of the data being queried.

See <https://www.upsolver.com/blog/apache-presto-what-is-it-and-why-you-should-use-it> (last accessed on November 10, 2023).

Upon information and belief, Chase's systems implement Presto:

See <https://www.linkedin.com/jobs/view/software-engineer-iii-python-sql-aws-at-jpmorgan-chase-co-3733025046/> (last accessed on November 10, 2023).

; see also

https://jpmc.fa.oraclecloud.com/hcmUI/CandidateExperience/en/sites/CX_1002/requisitions/preview/210456949/?keyword=presto&mode=location (last accessed Nov. 13, 2023), reproduced below:

	<p>Preferred qualifications, capabilities, and skills</p> <ul style="list-style-type: none"> • Experience providing technical leadership • Development experience writing applications using big data frameworks such as Yarn, MR, HDFS, Hive, and Presto <p>See https://www.simplyhired.com/job/VODNQqFgJUDpVPUqInujj_X7lax6F3VYdJpOso7yDx251PtHOwBiSw (last accessed on November 10, 2023).</p> <p>Upon information and belief, Chase’s systems implement Trino:</p> <p>See https://www.glassdoor.co.in/job-listing/senior-lead-software-engineer-snowflake-starburst-trino-j-p-morgan-JV_IC2865319_KO0.55_KE56.66.htm?jl=1008652600849 (last accessed on November 10, 2023).</p>
a virtual data warehouse including	<p>Upon information and belief, Chase’s systems include “a virtual data warehouse.” Chase implements Trino/Presto which allows Chase to handle data warehousing and analytics by querying large amounts of data where it lives.</p> <p>What Trino is</p> <p><u>Trino is a tool designed to efficiently query vast amounts of data using distributed queries.</u> If you work with terabytes or petabytes of data, you are likely using tools that interact with Hadoop and HDFS. Trino was designed as an alternative to tools that query HDFS using pipelines of MapReduce jobs, such as Hive or Pig, but Trino is not limited to accessing HDFS. Trino can be and has been extended to operate over different kinds of data sources, including traditional relational databases and other data sources such as Cassandra.</p> <p><u>Trino was designed to handle data warehousing and analytics: data analysis, aggregating large amounts of data and producing reports.</u> These workloads are often classified as Online Analytical Processing (OLAP).</p> <p>See https://trino.io/docs/current/overview/use-cases.html (last accessed on November 10, 2023).</p>

Trino allows users to perform data warehouse analytics without the data warehouse, enabling users to query data anywhere it lives. In practice, an analyst doesn't need to load the data, transform the data, or do preparation of any kind. With this, an analyst can then access data anywhere, using regular SQL queries, without having to worry about the underlying infrastructure that makes it all work. This is an alternative to the traditional approach of collecting and consolidating data in a centralized data warehouse.

See <https://www.work-bench.com/post/trino-a-distributed-query-engine> (last accessed on November 10, 2023).

What Presto Is

Presto is a tool designed to efficiently query vast amounts of data using distributed queries. If you work with terabytes or petabytes of data, you are likely using tools that interact with Hadoop and HDFS. Presto was designed as an alternative to tools that query HDFS using pipelines of MapReduce jobs such as Hive or Pig, but Presto is not limited to accessing HDFS. Presto can be and has been extended to operate over different kinds of data sources including traditional relational databases and other data sources such as Cassandra.

Presto was designed to handle data warehousing and analytics: data analysis, aggregating large amounts of data and producing reports. These workloads are often classified as Online Analytical Processing (OLAP).

See <http://prestodb.io/docs/0.283/overview/use-cases.html> (last accessed on November 10, 2023).

Presto's power and value proposition is defined by the fact that it can query data wherever it is stored, without the need to move the data into a separate, structured system like a relational database or data warehouse. Finally, query execution runs in parallel over a scalable "pure memory-based architecture," returning results in seconds irrespective of the size of the data being queried.

	<p>See https://www.upsolver.com/blog/apache-presto-what-is-it-and-why-you-should-use-it (last accessed on November 10, 2023).</p>
<p>a plurality of disparate back-end database systems associated with a common enterprise, each back-end database system capable of having a different legacy architecture and organization;</p>	<p>Upon information and belief, Chase's systems includes "a plurality of disparate back-end database systems associated with a common enterprise, each back-end database system capable of having a different legacy architecture and organization." Trino connects with multiple disparate databases such as MYSQL, MongoDB etc. Trino connects with a variety of different databases which have different architectures and organizations.</p> <p>Trino allows you to do that by using federated queries. A <i>federated query</i> is a SQL query that references and uses different databases and schemas from entirely different systems in the same statement. All the data sources in Trino are available for you to query at the same time, with the same SQL in the same query.</p> <p>See www.oreilly.com/library/view/trino-the-definitive/9781098107703/ch01.html (last accessed on November 10, 2023).</p> <p>Trino connects with multiple disparate databases such as MYSQL, MongoDB etc. Trino connects with a variety of different databases which have different architectures and organizations.</p> <p>Trino queries data where it lives and does not require a migration of data to a single location. So Trino allows you to query data in HDFS and other distributed object storage systems. It allows you to query RDBMSs and other data sources. As such, it can really query data wherever it lives and therefore be a replacement to the traditional, expensive, and heavy extract, transform, and load (ETL) processes. Or at a minimum, it can help you with them and lighten the load. So Trino is clearly not just another SQL-on-Hadoop solution.</p>

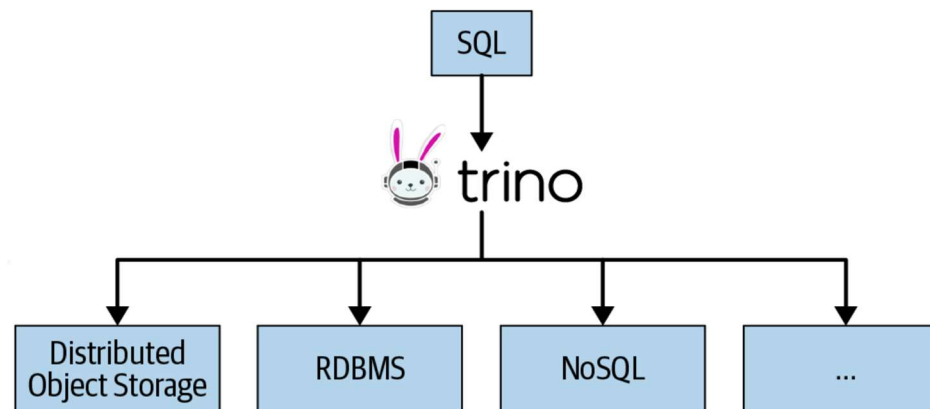


Figure 1-3. SQL support for a variety of data sources with Trino

Object storage systems include Amazon Web Services (AWS) Simple Storage Service (S3), Microsoft Azure Blob Stor-

See www.oreilly.com/library/view/trino-the-definitive/9781098107703/ch01.html (last accessed on November 10, 2023).

Presto connects with multiple disparate databases such as MYSQL, MongoDB etc. Presto connects with a variety of different databases which have different architectures and organizations.

Presto can query a single data source or a join across multiple data sources, such as AWS S3 and MySQL or Kafka and MongoDB. You can query traditional data sources and non-relational data sources (MongoDB, Elasticsearch) and data lakes (HDFS, AWS S3). This is one of the biggest advantages of Presto—you can query across disparate and different data models without having to move your data.

See www.ibm.com/downloads/cas/KBY4EKA0 (last accessed on November 10, 2023).

In other words, the user doesn't need to transport data into a separate, structured channel like a data warehouse or a relational database. In addition, PrestoDB uses a parallel architecture instead of a scalable, memory-based structure. This feature is particularly useful for enterprises as it allows PrestoDB to gather and analyze impressive amounts of data in mere seconds.

See <https://pandio.com/prestodb-drives-massive-value-for-enterprise-use/> (last accessed on November 10, 2023).

Accumulo Connector

BigQuery Connector

Cassandra Connector

ClickHouse connector

Druid Connector

Elasticsearch Connector

MongoDB Connector

MySQL Connector

Oracle Connector

PostgreSQL Connector

Prometheus Connector

Redis Connector

	See https://pandio.com/prestodb-drives-massive-value-for-enterprise-use/ (last accessed on November 10, 2023).
a database of record (DBOR) infrastructure for storing in a single physical location a plurality of metadata extracted from each back-end database system of the plurality of disparate back-end database systems	<p>Upon information and belief, Chase's systems include "a database of record (DBOR) infrastructure for storing in a single physical location a plurality of metadata extracted from each back-end database system of the plurality of disparate back-end database systems." Trino/Presto includes a catalog stored in the Trino/Presto configuration directory.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <h3>Catalog</h3> <p>A Trino catalog contains schemas and references a data source via a connector. For example, you can configure a JMX catalog to provide access to JMX information via the JMX connector. When you run SQL statements in Trino, you are running them against one or more catalogs. Other examples of catalogs include the Hive catalog to connect to a Hive data source.</p> <p>When addressing a table in Trino, the fully-qualified table name is always rooted in a catalog. For example, a fully-qualified table name of <code>hive.test_data.test</code> refers to the <code>test</code> table in the <code>test_data</code> schema in the <code>hive</code> catalog.</p> <p>Catalogs are defined in properties files stored in the Trino configuration directory.</p> </div>

Schema

Schemas are a way to organize tables. Together, a catalog and schema define a set of tables that can be queried. When accessing Hive or a relational database such as MySQL with Trino, a schema translates to the same concept in the target database. Other types of connectors may

See <https://trino.io/docs/current/overview/concepts.html> (last accessed on November 10, 2023).

Catalog

A Presto catalog contains schemas and references a data source via a connector. For example, you can configure a JMX catalog to provide access to JMX information via the JMX connector. When you run a SQL statement in Presto, you are running it against one or more catalogs. Other examples of catalogs include the Hive catalog to connect to a Hive data source.

When addressing a table in Presto, the fully-qualified table name is always rooted in a catalog. For example, a fully-qualified table name of `hive.test_data.test` would refer to the `test` table in the `test_data` schema in the `hive` catalog.

Catalogs are defined in properties files stored in the Presto configuration directory.

Schema

Schemas are a way to organize tables. Together, a catalog and schema define a set of tables that can be queried. When accessing Hive or a relational database such as MySQL with Presto, a schema translates to the

	<p>See http://prestodb.io/docs/0.283/overview/concepts.html (last accessed on November 10, 2023).</p>
<p>so as to form a logical enterprise data module, the logical enterprise data module responsive to information requests from the plurality of different client views.</p>	<p>Upon information and belief, Chase’s systems include a DBOR so as “to form a logical enterprise data module, the logical enterprise data module responsive to information requests from the plurality of different client views.”</p> <p>Trino’s/Presto’s coordinator provides a logical enterprise data module responsive to different queries.</p> <p style="text-align: center;">Coordinator</p> <p>The Trino coordinator is the server that is responsible for parsing statements, planning queries, and managing Trino worker nodes. It is the “brain” of a Trino installation and is also the node to which a client connects to submit statements for execution. Every Trino installation must have a Trino coordinator alongside one or more Trino workers. For development or testing purposes, a single instance of Trino can be configured to perform both roles.</p> <p>See https://trino.io/docs/current/overview/concepts.html (last accessed on November 10, 2023).</p>

A client is used to send queries to Trino and receive results, or otherwise interact with Trino and the connected data sources.

Some clients, such as the **command line interface**, can provide a user interface directly. Clients like the **JDBC driver**, provide a mechanism for other tools to connect to Trino.

See <https://trino.io/docs/current/client.html> (last accessed on November 10, 2023).

Coordinator

The Presto coordinator is the server that is responsible for parsing statements, planning queries, and managing Presto worker nodes. It is the “brain” of a Presto installation and is also the node to which a client connects to submit statements for execution. Every Presto installation must have a Presto coordinator alongside one or more Presto workers. For development or testing purposes, a single instance of Presto can be configured to perform both roles.

See <http://prestodb.io/docs/0.283/overview/concepts.html> (last accessed on November 10, 2023).

Trino’s/ Presto’s coordinator provides a logical enterprise data module responsive to different queries.

When the Presto coordinator SQL server receives an SQL query from a user, the first thing it does is use a custom query to parse, plan and schedule a distributed plan across the other nodes. Presto Rest API is used to submit query statements for execution on a server and to retrieve the results for the client. Presto supports standard ANSI SQL meanings, including joins, queries, sub-queries and aggregations. Once it has compiled the query, Presto parses the request into different stages between the worker nodes.

Since Presto was built on the concept of data abstraction, it is extensible to any data source and can easily query data sources such as data lakes, data



Reporting

Presto allows for data to be queried from multiple sources generating a single, easily accessible report or dashboard for BI purposes. Presto is simple and easy enough to use that analysts can conduct queries and create reports without the help of engineers.



Analytics

Presto enables analysts to conduct queries on both structured and unstructured data directly on a data lake without going through a data transformation process.

See <https://www.ibm.com/topics/presto> (last accessed on November 10, 2023).

